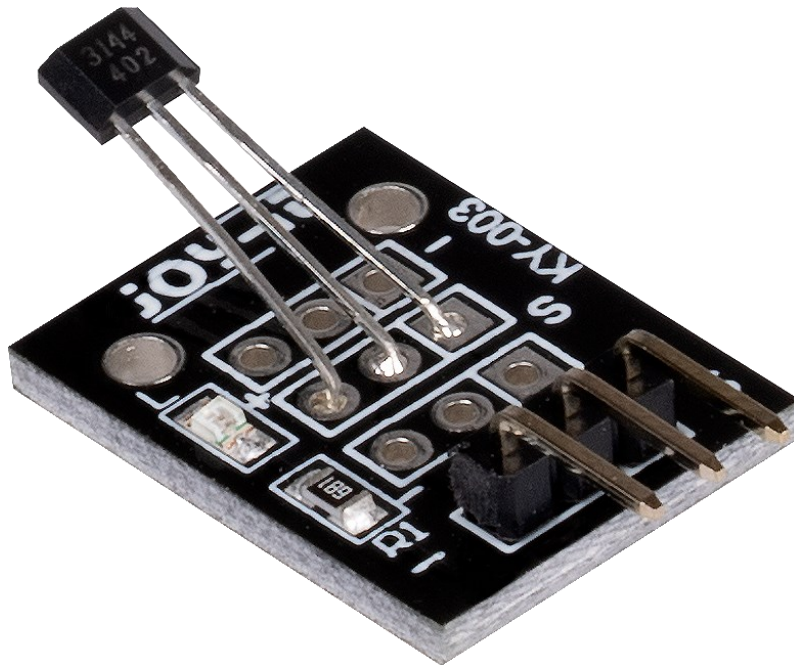


HALL MAGNETFELDESENSOR



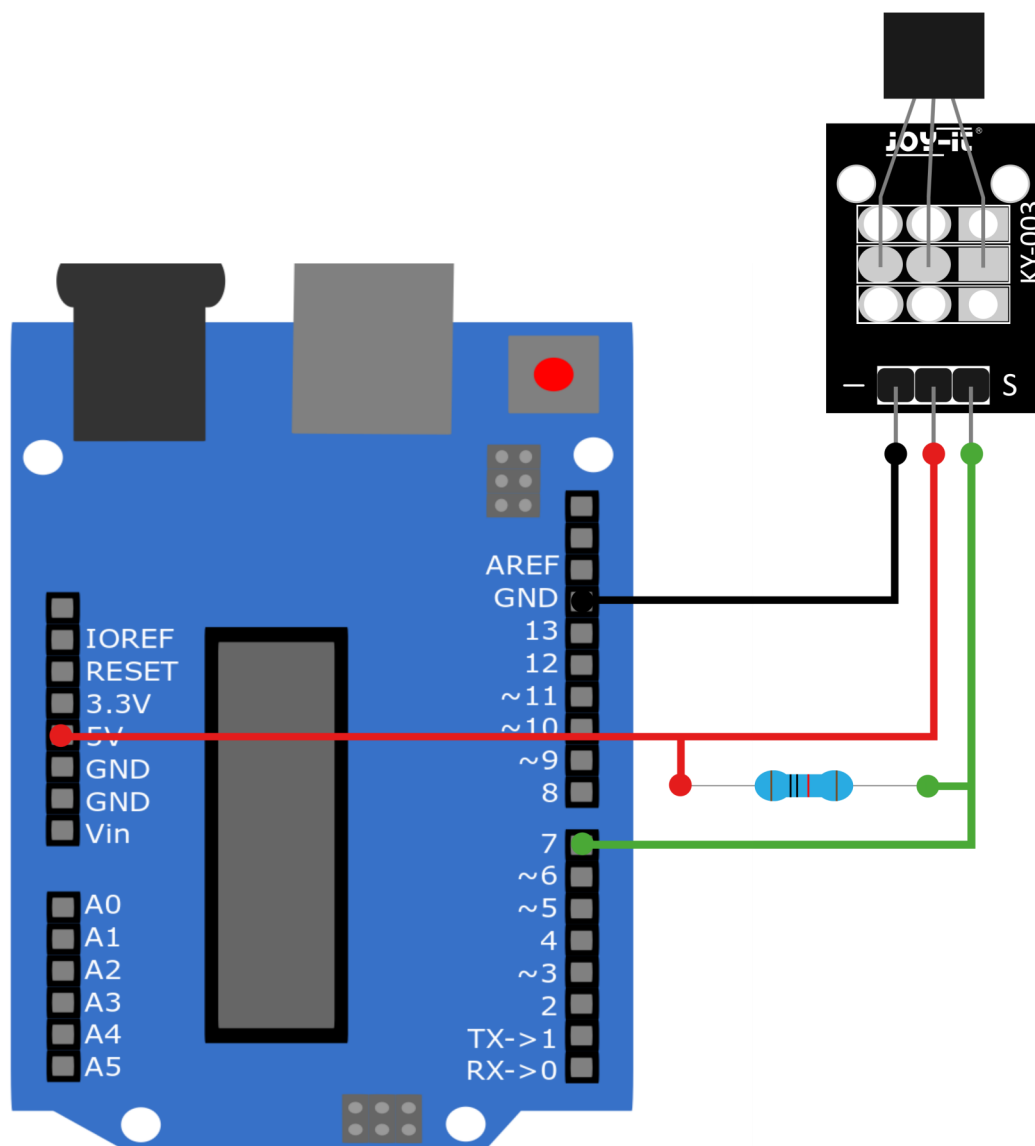
1. ALLGEMEINE INFORMATIONEN

Sehr geehrte/r Kunde/in,
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

2. VERWENDUNG MIT DEM ARDUINO

Schließen Sie Ihren Sensor, wie im Schaubild und der Tabelle unten zusehen, an Ihren Arduino an. Für optimale Ergebnisse sollten Sie einen Pullup Widerstand (ca. 10 k Ω) zwischen der Signalleitung des Sensors und der 5 V Leitung verwenden.



Arduino	Sensor
GND	GND -
5 V	VCC +
Pin 7	Signal

Codebeispiel

In dem nun folgenden Codebeispiel, wird jede Sekunde im seriellen Monitor angegeben, ob ein Magnetfeld erkannt wurde. Kopieren Sie den Beispielcode und laden Sie ihn auf Ihren Arduino hoch. Achten Sie darauf, dass Sie das richtige Board und den richtigen Port in Ihrer Arduino IDE ausgewählt haben.

```
// Deklaration und Initialisierung des Eingang-Pins

int Digital_Eingang = 7; // Digital-Pin

void setup ()
{
  pinMode (Digital_Eingang, INPUT);

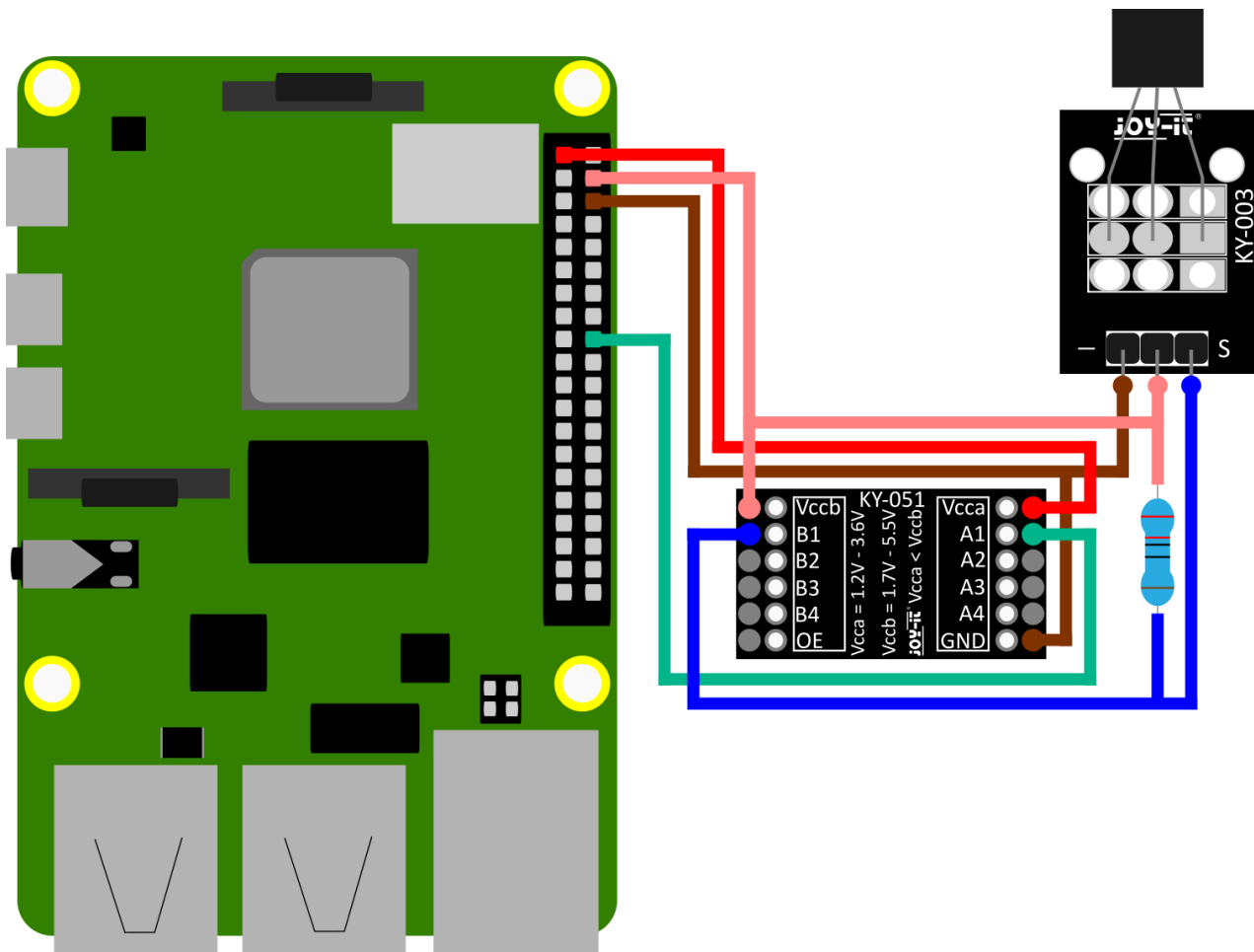
  Serial.begin (9600); // Serielle Ausgabe mit 9600 bps
}

// Das Programm liest den aktuellen Werte des Eingang-Pins
// und gibt diese auf der seriellen Ausgabe aus
void loop ()
{
  int Digital;

  //Aktueller Wert wird ausgelesen
  Digital = digitalRead (Digital_Eingang);
  Serial.print ("Magnetfeld");
  if(Digital==1)
  {
    Serial.println (" nicht erkannt");
  }
  else
  {
    Serial.println (" erkannt");
  }
  Serial.println ("-----");
  delay (1000);
}
```

3. VERWENDUNG MIT DEM RASPBERRY PI

Schließen Sie Ihren Sensor, wie im Schaubild und der Tabelle zusehen an einen Voltage Translator und Ihren Raspberry Pi an. Hierbei empfehlen wir den [KY-051 Voltage Translator von Joy-IT](#). Für optimale Ergebnisse sollten Sie einen Pullup Widerstand (ca. 10 k Ω) zwischen der Signalleitung des Sensors und der 5V Leitung verwenden.



Raspberry Pi	Sensor	Voltage Translator
-	Signal	B1
5V (Pin 4)	+V	
Masse (Pin 6)	GND	GND
3,3V (Pin 1)	-	Vccb
5V (Pin 4)	-	Vcca
GPIO 24 (Pin 18)	-	A1

Codebeispiel

Erstellen Sie auf Ihrem Raspberry Pi nun eine neue Datei mit dem folgenden Befehl:

```
sudo nano SEN-KY003HMS.py
```

Kopieren Sie das nun folgende Codebeispiel in den Editor. Speichern Sie dann die Datei mit der Tastenkombination STRG+O und verlassen Sie die Datei mit der Kombination STRG+X.

```
import time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

delayTime = 1      #Zeit zwischen den Messungen
Digital_PIN = 24   #Digital-Pin (BCM)
GPIO.setup(Digital_PIN, GPIO.IN)
while True:
    # Ausgabe
    if GPIO.input(Digital_PIN) == False:
        print ("Magnetfeld erkannt!!")
    else:
        print ("Kein Magnetfeld erkannt")
    print ("-----")

    time.sleep(delayTime)
```

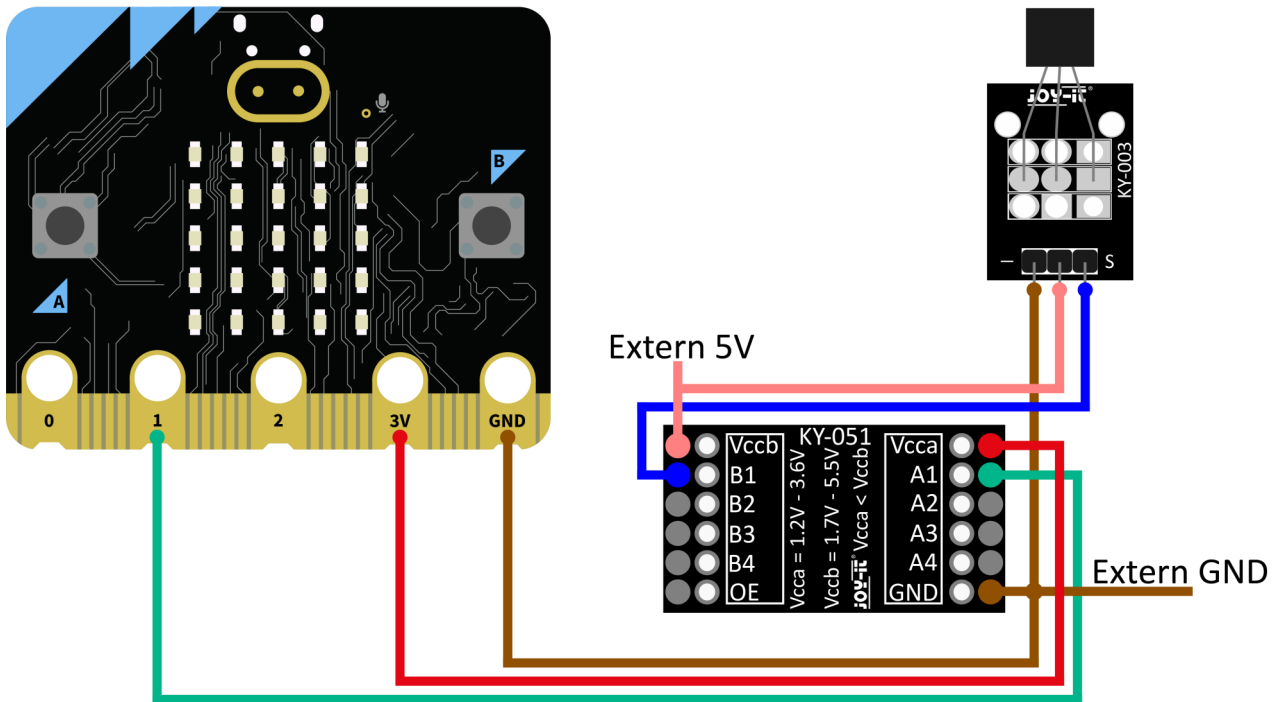
Starten Sie nun das Programm mit dem folgenden Befehl:

```
sudo python3 SEN-KY003HMS.py
```

In der Konsole wird Ihnen nun angezeigt, ob ein magnetisches Feld erkannt wurde.

4. VERWENDUNG MIT DEM MICRO:BIT

Schließen Sie Ihren Sensor, wie im Schaubild und der Tabelle zusehen an einen Voltage Translator und Ihren Micro:Bit an. Hierbei empfehlen wir den [KY-051 Voltage Translator von Joy-IT](#). Für optimale Ergebnisse sollten Sie einen Pullup Widerstand (ca. 10 kΩ) zwischen der Signalleitung des Sensors und der 5V Leitung verwenden.



KY-003	Voltage translator
GND	Extern GND
+	[Extern 5 V]
Signal (S)	B1

Micro:Bit	Extern
GND	Extern GND

Voltage translator	Extern
Vccb	Extern 5 V
GND	Extern GND

Micro:Bit	Voltage Translator
Pin 1	A1
3 V	Vcca

Codebeispiel

In dem nun folgenden Codebeispiel, wird jede Sekunde im seriellen Monitor angegeben, ob ein Magnetfeld erkannt wurde. Sie können das Codebeispiel entweder in Ihrem MakeCode-Sketch nachbauen, oder die [.hex-Datei](#) herunterladen, und als MakeCode-Projekt importieren.

```

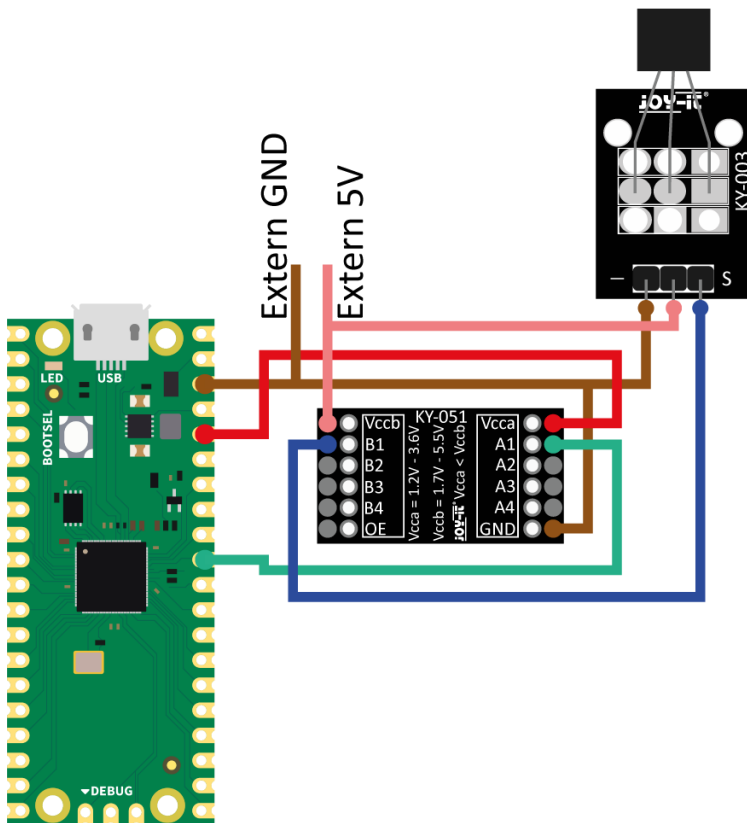
dauerhaft
  seriell Zeile ausgeben analoge Werte von Pin P1
  wenn analoge Werte von Pin P1 < 40 dann
    seriell Zeile ausgeben "Magnetfeld"
  ansonsten
    seriell Zeile ausgeben "Kein Magnetfeld"
  seriell Zeile ausgeben " "
  pausiere (ms) 1000

```

The image shows a Scratch code editor snippet for a MakeCode sketch. It starts with a 'dauerhaft' (forever) loop block. Inside the loop, there are several blocks: a 'seriell Zeile ausgeben' (serial print) block with the text 'analoge Werte von Pin P1', a 'wenn' (if) block with the condition 'analoge Werte von Pin P1 < 40', a 'dann' (then) block containing a 'seriell Zeile ausgeben' block with the text 'Magnetfeld', an 'ansonsten' (else) block containing a 'seriell Zeile ausgeben' block with the text 'Kein Magnetfeld', a 'seriell Zeile ausgeben' block with a single space character ' ', and finally a 'pausiere (ms)' (wait) block with the value '1000'.

5. VERWENDUNG MIT DEM RASPBERRY PI PICO

Schließen Sie Ihren Sensor, wie im Schaubild und der Tabelle zusehen an einen Voltage Translator und Ihren Raspberry Pi Pico an. Hierbei empfehlen wir den [KY-051 Voltage Translator von Joy-IT](#). Für optimale Ergebnisse sollten Sie einen Pullup Widerstand (ca. 10 kΩ) zwischen der Signalleitung des Sensors und der 5V Leitung verwenden.



KY-003	Voltage translator
GND	Extern GND
+	[Extern 5 V]
Signal (S)	B1

Raspberry Pi Pico	Extern
GND	Extern GND

Voltage translator	Extern
Vccb	Extern 5 V
GND	Extern GND

Raspberry Pi Pico	Voltage Translator
GPIO26	A1
3 V	Vcca

Codebeispiel

In dem nun folgenden Codebeispiel, wird jede 0,5 Sekunden in der Konsole angegeben, ob ein Magnetfeld erkannt wurde. Kopieren Sie den Beispielcode und laden Sie ihn auf Ihren Raspberry Pi. Alternativ können Sie den Beispielcode auch [hier](#) herunterladen.

```
# Bibliotheken Laden

from machine import Pin, Timer

from time import sleep

# Initialisierung von GPIO als Eingang

sensor = Pin(26, Pin.IN, Pin.PULL_DOWN)

# Dauerschleife für kontinuierliche Serielle Ausgabe

while True:

    if sensor.value() == 0:

        print("Magnetfeld")

    else:

        print("Kein Magnetfeld")

    print("-----")

    sleep(0.5)
```

6. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)



Symbol auf Elektro- und Elektronikgeräten:

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

Möglichkeit Rückgabe in Ihrer Nähe:

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an Service@joy-it.net oder per Telefon an uns.

7. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 98469 – 66 (10 - 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net